International Journal of Information and Computer Science
IJICS Volume 1, Issue 6, September 2012 PP. 144-152
ISSN(online) 2161-5381
ISSN(print) 2161-6450
www.iji-cs.org

High-Performance Post-Placement Length Estimation Techniques

Yangyang Li[1], Amin Farshidi[1], Laleh Behjat[1], William Swartz[2]
[1]Department of Electrical and Computer Engineering, University of Calgary, Calgary, Canada
[2]InternetCAD.com, Inc. Dallas, Texas, US
Email: yangyli@ucalgary.ca

(Abstract) Recently there has been large emphasis on congestion based placement where the quality of placement is measured using the global routing results. However, during the placement process, only half perimeter wire length (HPWL) estimation technique which provides a very simplistic calculation is used. In this paper, we first show very important factors that are ignored by HPWL and can affect wire length, routability and congestion in the routing stage. Then, we show that length estimation with high accuracy and fidelity and low cost is feasible on the most recent routability-driven placement benchmarks. In addition, we propose techniques that can be used during the placement stage to improve the global and detailed routing results.

Keywords: Wire Length Prediction; Congestion; Placement; Routing.

1. INTRODUCTION

The most commonly used estimation technique for measuring the quality of a placement is half perimeter wire length (HPWL) estimation. HPWL produces a rough estimate versus a tight lower bound of the wire length, but it can be computed with low runtime.

As the circuits are made with higher density, total HPWL does not provide a good measure for the routability of a circuit. Hence, metrics such as congestion are considered during placement to improve the final results. For example, placers in ISPD2011 Routability-Driven Placement Contest [1], incorporated congestion as part of the objective or cost of a placement. However, wire length still remains an important objective that cannot be neglected [2].

This is demonstrated by the fact that in many of the successful placers of the ISPD2011 contest, such as [2,3,4], the main objective is reducing the wire length and provisions for routability are added either to the objective or the constraints. The numerical simulations performed in this paper show that the HPWL can be up to 70% lower than the routed wire length, and up to 20% lower than the routed wire length for the nets with low degrees. Hence, using HPWL to optimize a placement can result in solving the wrong optimization problem and consequently many more problems during the routing stages.

In this paper, several characteristics of a high-performance length estimation technique are proposed. The main properties considered when designing these techniques are:

- How close the results are to the global routing results?
- How low is the runtime?
- Is the technique able to handle 3D designs?
- Are there any considerations for detailed routing?

High Quality Length Estimates: We propose several ways to achieve high quality length estimates at low cost during the placement process. First, a set of required properties for a technique that can produce more accurate results is defined. Then, the properties of the existing estimation techniques are studied to see where the source of their shortcomings are and modifications to the HPWL estimates that can increase the accuracy of the estimates are proposed. Finally, considerations for detailed routing are discussed.

The rest of this paper is organized as follows: In Section 2, a background on wire length estimation along with the motivations of this work are presented. The shortcomings of HPWL estimation are discussed and the performance of HPWL and Flute are compared under several conditions and for different criteria in Section 3. In Section 4, the necessary considerations during placement to improve the detailed routing performance are proposed and the relevant experimental results are presented. Finally, conclusions and future work are discussed in Section 5.

2. PRELIMINARIES AND MOTIVATIONS

Placement and routing are two main stages of VLSI physical design. Traditionally, during placement, the location of the circuit components, cells, are determined. During the routing stage, these locations are normally considered as fixed and are used to find paths of the wires [5]. In traditional physical design processes, placement and routing are performed separately and sequentially, therefore the quality of routing highly depends on how the placement solution is obtained [6]. If a highly congested area exits after placement, the routing will become very difficult to

solve, and many wires will have lengths that will be much higher than anticipated. In [7], challenges in routing due to the increasing number of metal layers, design rules and design complexity are discussed. Because of the significance of these challenges, a congestion-based placement competition was held in ISPD2011.

HPWL is a fast and low cost technique to estimate the after placement wire length. In [8], the first report on the discrepancies between HPWL calculations for different placers is presented. The paper showed vast differences on how HPWL was calculated and resulted in standardization of HPWL calculation during placement. Other commonly used techniques to estimate post placement wire length are BI1ST [9] and Flute [10]. BI1ST approximates the length using only one Steiner point. In Flute, an approximation for RSMT for each net is calculated using a lookup table. HPWL and BI1ST calculate the exact minimum lengths for degree-two and degree-three nets. Flute is capable of finding the RSMT for nets with degree up to nine.

Several researchers have tried to develop techniques to improve the quality of wire length estimates. In [11], an empirical model for probabilistic wire length estimation for each net in a placed circuit is proposed. For longer nets, the error of the HPWL estimation increases rapidly, as illustrated in [12], when compared with rectilinear Steiner minimal tree (RSMT) evaluators. However, rectilinear Steiner minimal tree generation is an NP-hard problem and the cost of producing this type of trees is prohibitive.

The effects of congestion on the global routing results are studied in [13]. A technique is then proposed to identify the Groups of Tangled Logic (GTL) which are the highly congested areas in the placement and may negatively affect routing results. Finally, remedial actions are presented which are shown to be effective in balancing the congestion in placement area.

Routability-driven placement has other sets of problems such as via locations. As in the global routing stage only bends are counted as vias, the extra vias needed for horizontal and L-shaped nets are neglected. Several solutions for this problem have been proposed. In [14], this problem is mentioned and an effective congestion metric which considers both the via capacity and local nets in global routing is proposed. Some 3D routers [15] consider reducing the number of vias as an important objective function.

Several researchers have developed pre-placement length estimation techniques e.g. [16, 17, 18, 19], which is a much harder problem. Net degree has been used by many researchers as an indicator for estimating the net lengths before placement, with the general consensus that the higher the net degree the longer the net. Other metrics to measure the quality of estimation are absolute error or mean squared error. However, none of these metrics provides information on how the estimates follow the trends of the actual data. Fidelity, which is a measure that shows how correlated the ordering of the estimates and the ordering of the actual data are, should be analyzed to better prove the effectiveness of the model [20]. Therefore, in [18, 19], correlation coefficient is employed to validate the accuracy of the wire length estimates.

MOTIVATIONS: Since [8], there has been a huge improvement in placement and routing algorithms. However, there has been no improvement in how the quality of a placement is measured during placement. In this paper, we try to close the gap by determining where most improvement can be achieved with the least cost. Also, issues that are related to modern circuits, such as via location and local nets, are discussed and algorithms to account for these are developed.

3. INCREASING ESTIMATES ACCURACY

HPWL estimates have been used for decades to measure the quality of a placement. However, the estimates are not accurate when compared to the routed wire lengths and in this paper we show how more accurate estimates can be obtained at low cost.

One technique to increase the accuracy of length estimation during placement is to use actual routes versus bounding box for calculating the length estimates. Flute [10], can produce routes for nets with high accuracy in low runtime. In **Figure 1**, the HPWL estimates, and the estimates obtained from Flute are compared to after routing estimates for the ISPD2011 benchmark circuits [1]. In this figure, the x-axis represents the circuit name and the y-axis shows the percentage ratio of the HPWL or Flute estimates over the routed wire length. From this figure, it can be seen that HPWL is on average 13% and in worse case, 30% less than the routed wire length. In comparison, the results obtained by Flute are on average 2.7% and in worse case only 4% lower than the routed wire length.

In the experiments of **Figure 1**, mPL11 [1] from ISPD2011 contest is used for placement and GRIP [15] is used for routing. However, the results for other placers in the contest are similar to mPL11 results and are given in **Table 1**. It should be mentioned that in this table only placers that were available to the authors were used. In this table, Columns 2 and 3 include the number of nets and cells

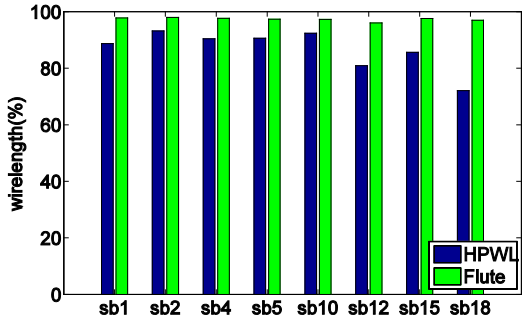


Figure 1: Comparison between HPWL, Flute and routed wire length for ISPD11 benchmark circuit.

Table 1: Circuit Statistics, wire length and runtime results comparing HPWL, and Flute with the routed wire length for three placers on the ISPD 2011 benchmark suite.

Circuit	# nets	# cells	% ratio to total routed length						ratio to placement time (×10^{-3})					
			mPL11		SimPLR		Ripple		mPL11		SimPLR		Ripple	
			hpwl	Flute	hpwl	Flute	hpwl	Flute	hpwl	Flute	hpwl	Flute	hpwl	Flute
sb1	822,744	847,441	88.7	97.8	87.1	98.0	87.9	98.0	1.92	6.71	1.21	4.16	0.69	2.40
sb2	990,899	1,014,029	93.2	98.0	91.7	97.8	91.3	97.7	1.92	5.01	1.06	2.74	0.49	1.27
sb4	567,607	600,220	90.4	97.7	87.8	97.6	88.2	97.7	1.90	5.77	1.53	4.87	0.62	1.96
sb5	786,999	772,457	90.6	97.3	89.9	97.6	90.4	97.6	1.72	4.98	1.15	3.40	0.57	1.76
sb10	1,085,737	1,129,144	92.4	97.3	92.3	98.1	92.5	97.9	1.78	5.70	0.69	2.26	0.45	1.32
sb12	1,293,436	1,293,433	80.9	96.0	77.3	97.3	77.6	97.1	1.57	6.62	1.09	4.61	0.52	2.19
sb15	1,080,409	1,123,963	85.6	97.6	85.8	98.2	86.0	97.9	1.75	7.42	1.17	4.92	0.51	3.35
sb18	468,918	483,452	72.1	97.0	67.7	97.3	73.2	96.9	1.65	10.22	1.40	8.99	0.35	2.22
Ave.	-	-	86.7	97.3	84.9	97.7	85.9	97.6	1.78	6.55	1.16	4.50	0.53	2.06

in the ISPD2011 circuits. In Columns 4 to 9, the percentage ratio of the length estimates over the routed wire length when the estimates are found by HPWL, or Flute for three different placers, mPL11, SimPLR [2] and Ripple [3], are given. In Columns 10 to 15, the ratio of the runtime of the HPWL and Flute to the total placement time is given. As it can be seen this table, the accuracy of the length estimates is greatly improved when Flute is used to estimate the length, but the cost on the runtime is negligible, as on average, around 8,000 seconds is spent on placement of a circuit while Flute spends only 30 seconds on average for calculating the wire length estimates. In addition, the initial routing obtained using Flute can be used in the global routing step.

The fidelity [20] of the above techniques for estimating the lengths of the low-degree nets is calculated to ensure that the high performance of Flute is reliable and where possible shortcomings come from. Fidelity is a measure that shows how correlated the ordering of the estimates and the ordering of the actual data are. Correlation coefficient is one of the most commonly used fidelity metrics [21] and is employed in this paper to validate the fidelity. The correlation coefficient is calculated using the following equation:

$$CorrCoeff(\vec{L}_{est}, \vec{L}_r) = \frac{COV((\vec{L}_{est}, \vec{L}_r))}{\sigma_{\vec{L}_{est}} \sigma_{\vec{L}_r}}$$

Where, $\vec{L}_{est}$ and $\vec{L}_r$ represent the vectors of all net length estimates and the actual routed wire lengths, respectively. $COV(\vec{L}_{est}, \vec{L}_r)$ is the covariance of the estimated and routed wire lengths. $\sigma_{\vec{L}_{est}}$ and $\sigma_{\vec{L}_r}$ are the standard deviations of the estimated and routed wire lengths, respectively.

The fidelity analysis results are presented in **Table 2**. In Columns 2 and 3, the correlation coefficients of the routed wire length to the wire length estimated by HPWL and Flute are presented, respectively, when the circuit is placed by mPL11 and routed using GRIP. The same results are presented for SimPLR and Ripple placers in Columns 4 and 5 and Columns 6 and 7, respectively. It can be seen that the results obtained by Flute are highly correlated to the after routing results, but HPWL estimates show a relatively lower fidelity which makes the results less reliable.

Table 2: Correlation coefficients of the length estimates to the routed wire length for the ISPD 2011 benchmark suite benchmark are considered

Circuit	Total Correlation (%)						Degree 2-10 Correlation (%)						Degree >100 Correlation (%)					
	mPL11		SimPLR		Ripple		mPL11		SimPLR		Ripple		mPL11		SimPLR		Ripple	
	hpwl	Flute	hpwl	Flute	hpwl	Flute	hpwl	Flute	hpwl	Flute	hpwl	Flute	hpwl	Flute	hpwl	Flute	hpwl	Flute
sb1	85.9	99.1	85.3	99.4	87.9	99.3	99.8	99.9	99.6	99.9	99.7	99.9	74.9	95.4	71.9	96.9	66.5	95.9
sb2	96.9	99.8	95.6	99.9	95.6	99.9	99.7	99.8	99.5	99.8	99.5	99.8	88.7	99.8	80.2	99.9	86.9	99.9
sb4	92.6	99.8	90.8	99.8	91.1	99.9	99.8	99.9	99.7	99.8	99.8	99.9	81.0	99.6	84.3	99.5	81.0	99.6
sb5	94.3	98.6	95.4	99.4	95.7	99.4	99.4	99.8	99.4	99.8	99.5	99.9	74.5	75.6	76.1	85.9	78.2	84.4
sb10	96.3	98.8	95.7	98.9	96.3	99.1	99.4	99.4	99.7	99.8	99.7	99.8	82.3	83.6	77.8	80.2	82.4	84.7
sb12	81.8	94.4	89.8	98.0	88.5	98.2	98.6	98.2	98. 9	99.7	99.9	99.8	80.5	81.1	71.2	84.9	64.7	82.8
sb15	88.8	99.8	89.5	99.8	90.1	99.8	99.5	99.8	98.6	99.9	99.6	99.9	9.9	99.4	18.7	99.5	16.0	99.5
sb18	81.0	99.1	81.8	99.3	85.7	99.3	99.4	99.7	99.3	99.6	99.3	99.7	93.3	96.5	94.2	97.2	91.3	96.2
Avg.	89.7	98.7	90.5	99.3	91.4	99.4	99.5	99.7	99.5	99.8	99.5	99.8	73.1	91.4	71.8	93.0	70.9	92.9

The accuracy of the length estimation and hence the possible performance of the placement is greatly increase

the accuracy of the length estimation and hence the possible performance of the placement is greatly increased by using Flute as an estimator. However, several questions remain that need to be considered and answered such as:

- Where does the difference between the estimates and the routed wire lengths come from?
- How can vias be considered during placement?
- Can there be considerations added for the detailed routing stage?

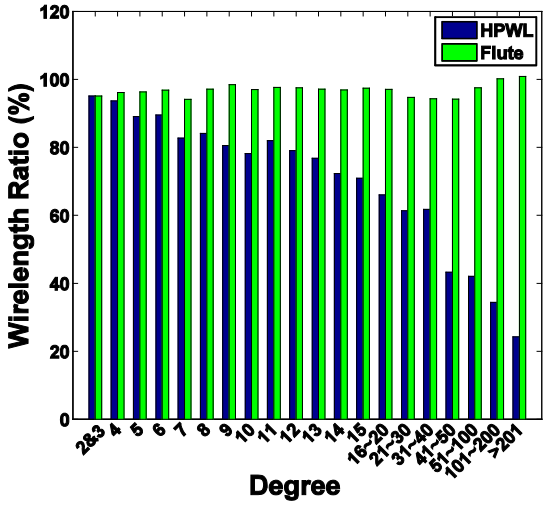
In the rest of this paper, each one of these questions is addressed and appropriate techniques are proposed to improve length estimation.

3.1. Degree-Based Estimation

In this paper, net degree is used to differentiate between the techniques used for estimating the lengths of nets, and it is shown that the origin of the error for low-degree net length estimates is very different from the high-degree nets. Therefore, to build a successful placement algorithm, these differences should be considered.

To find out the origin of the error, HPWL and Flute estimation results are compared based on the degrees of nets. In **Figure 2**, the percentage ratio of HPWL and Flute estimates over the routed wire lengths are shown for different of net degrees. In this experiment, all nets of Superblue12 which is the largest circuit in the ISPD2011 benchmark are used. In this figure, the x-axis is the degree of the nets and the y-axis shows the percentage of the ratio of the HPWL or Flute estimates to the routed wire length. From this figure, it can be seen that the accuracy of HPWL declines sharply as the degree of the nets increases, but the accuracy of Flute does not change significantly based on the

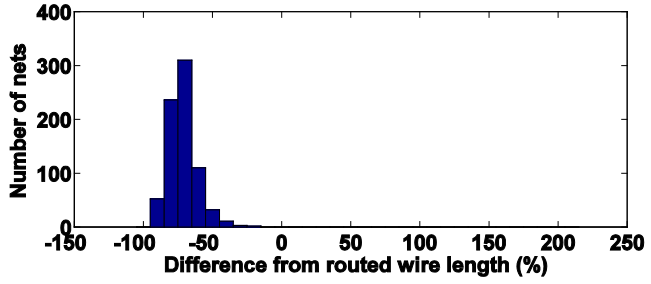
Figure 2: Comparison of HPWL and Flute estimates versus the routed wire lengths for nets with different degrees, when all nets



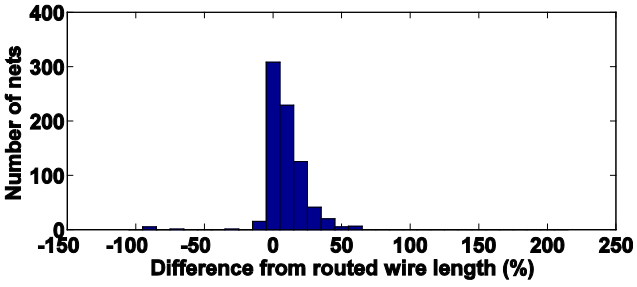
in SuperBlue12 from ISPD2011 benchmark are considered.

net degree. The same trend can be seen in Columns 8 to 19 of **Table 2**. In Columns 8 to 13 of this table, correlation coefficient between estimated and routed wire lengths for the three considered placers for nets with degree 2 and 3 is given. As it can be seen from these columns, the routed and estimated lengths are highly correlated. However, in columns 14 to 19, the difference between the performance of HPWL and Flute can be seen for high degree nets, where for example, the correlation between HPWL and routed lengths can be as low as 10%, i.e. virtually not correlated, for nets with degree higher than 100. The results indicate that HPWL estimates for high-degree nets are not reliable when compared to Flute estimates.

Most of the difference between the HPWL estimates and the routed wire length is due to the fact that HPWL ignores the connections inside the bounding box for high degree nets. The difference between the performance of Flute and HPWL can be seen through the histograms in **Figures 3(a**) and **3(b)**, where the percentage difference between the routed wire length and the estimates for nets with degree 100 and higher are given for HPWL and Flute, respectively. In this figure, zero means that there is no difference between the routed wire length and the estimate, positive value means that the routed wire length is shorter than the estimate, and negative means that the routed wire length is longer than the estimate. From these histograms, it can be seen that HPWL consistently underestimates the lengths by a large margin with the average net length being 70% lower than actual length. On the other hand, Flute performs very well on estimating the lengths of high-degree nets. The reason for over-estimation is because of inaccuracy of the routed lengths and will be discussed the following paragraphs.



(a) HPWL



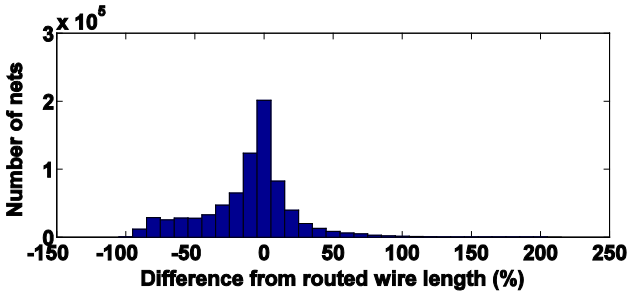
(b) Flute

Figure 3: Percentage difference between HPWL and Flute length estimates and the routed wire length for nets with high degree.

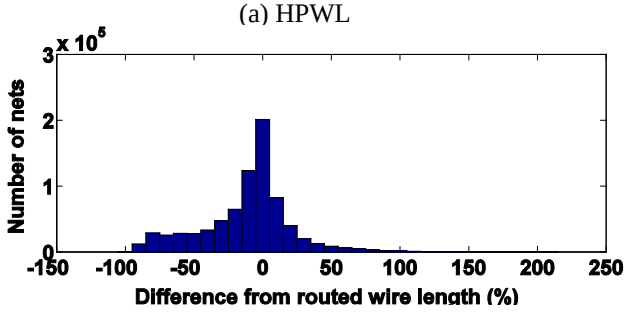
The situation is very different for nets with lower degrees (degree-two and degree-three). In **Figures 4(a)** and **4(b)**,

histograms for the percentage difference between the routed wire length and the estimates for nets with degree-two and degree-three are given for HPWL and Flute, respectively. From these figures, it can be seen that both HPWL and Flute might either over-estimate or under-estimate the wire lengths.

The explanation for higher routed lengths versus HPWL or Flute lengths is quite intuitive. The situation happens when there is a large amount of congestion that run horizontally or vertically in a circuit and a route has to make large detours to be able to connect the pins of a net. An example of such routes is given in **Figure 5** where the pin density and the routing path of degree-two nets with



(a) HPWL



(b) Flute

Figure 4: Percentage difference between HPWL and Flute length estimates and the routed wire length for nets with low degree.

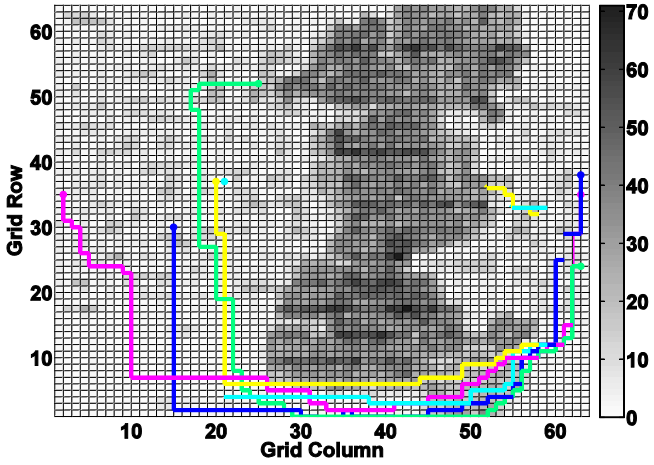


Figure 5: Pin density and routes of the outlier nets.

high length increase are shown for circuit IBM01 from ICCAD04 benchmark suite [22]. In this figure, four outlier nets whose pins are located in two sides of a congested area are shown. These nets need to make large detours to be able to make the right connections. It should be mentioned that routing algorithms based on maze routing tend to have a larger number of outliers.

Over-estimating of HPWL can be very surprising as HPWL is a lower bound of the length of the nets and hence should not be higher than the routed lengths. The error is because of how routed wire lengths are calculated. In the HPWL calculation, the exact locations of pins are considered. However, in global routing, all the pins are assumed to be located at the center of grid cells in the routing grid. Hence, if the pins of two nets are at the opposite boundaries of two routing grids, the routed length will be shorter than the HPWL.

3.2. 3D Length Estimation

Normally in calculating the HPWL, layer assignment and number of vias are not considered. However, vias make a big contribution in the total wire length and more importantly in the routability of a circuit. For every bend in the route of the circuit, at least two units of length (one for going up to the next layer and one for coming back) are added to the total length of the net.

These extra vias are ignored by all estimation techniques and hence by the placers that use them, but they not only contribute to the total wire length, but also are important factors in the routability of a placement. In **Table 3**, the percentage of the L-shaped degree-two nets for the ISPD2011 circuits and the minimum number of vias that will be added because of these nets are given for three of the top placers in the ISPD2011 contest. It can be seen that on average more than 90% of degree-two nets are L-shaped which means that over a million vias have to be accommodated during routing. The gravity of this situation can be alleviated if relevant penalties are added during placement and length estimation to compensate for the extra

Table 3: Percentage of L-shaped nets in ISPD2011 circuits

Cir.	% L-shaped degree-two nets					
	mPL11		SimPLR		Ripple	
	%	min vias	%	min vias	%	min vias
sb1	87	929164	93	989502	91	973762
sb2	88	1157184	95	1247534	94	1243270
sb4	85	683098	95	762664	92	741220
sb5	92	983678	96	1027536	95	1019150
sb10	93	1352672	96	1392434	95	1381660
sb12	94	1366194	96	1394364	95	1384110
sb15	87	1474652	94	1590730	92	1565498
sb18	79	503012	97	618642	96	612030
Ave.	88	1056207	95	1127926	94	1115088

lengths of the L-shaped wires.

4. DETAILED ROUTING CONSIDERATIONS

4.1. Overflow Calculations

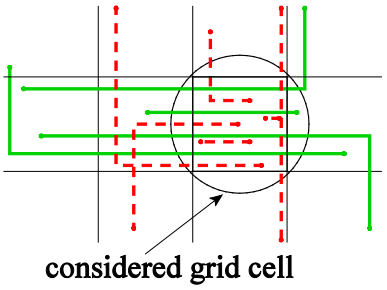
Since ISPD2011 placement contest, the placement objective has shifted from reducing the total wire length to routability. However, the only net segments that are considered for routability are the nets that are mapped to the global routing grid and no consideration is given to the nets that are entirely or partially inside a grid cell and are eliminated during global routing. Hence, feasibility problems can arise during detailed routing where a seemingly good placement and global routing solution becomes infeasible when ignored net segments are visited.

Another major problem of not considering the detailed density during global routing is that most of the removed nets are in congested areas which are usually harder to deal with. By ignoring the effects of these local nets, the difficulty of performing detailed routing is significantly increased. In [23], methods of modeling local nets based on Steiner tree wire length and pin densities are proposed.

In this paper, an algorithm with linear time complexity is given that can give an accurate measure for the routing density of each grid cell and hence the entire circuit. The routing density of a grid cell is defined as the number of net segments, horizontal or vertical, that are inside the grid cell. During the detailed routing, the maximum number of horizontal or vertical segments is of interest as it decides if a solution is routable or not. Using the proposed algorithm, this maximum can be calculated at low cost and hence the routability of the circuit can be ensured.

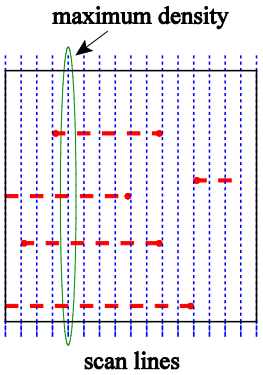
The density of each routing grid cell is first separated into two components, global density, D_G, and detailed density D_L. The global component is a non-negative integer equal to the number of segments which cross through the grid cell. The detailed component accounts for any partial segments entirely inside a grid cell. Furthermore, because the routing is performed in horizontal and vertical layers, each layer can be dealt with separately to reduce the complexity of the algorithm. An example of the global and detailed densities is shown in **Figure 6**. In **Figure 6(a)**, a routing solution along with the grid cells is shown. The global components are shown with solid lines and the detailed components are shown with dashed lines.

The key observation in the proposed algorithm is that the density can only change at pin or Steiner point locations, assuming a Steiner tree decomposition, and using this observation, in the proposed algorithm, *Density Cal.* given in **Figure 7**, first each net is decomposed into a set of horizontal and vertical segments using a Steiner tree decomposition, and for each grid cell, g, all net segments that are in a direction, horizontal or vertical, are put in the set $Segs(g)$. The global, D_G, and detail, D_L, densities of the grid cell are initialized to zero and an empty interval



(a) A routing solution

(b) Global routing components



(c) Detailed routing components

Figure 6: An example of a grid cell and its global and detailed components.

Figure 7: Algorithm for accurately computing routing density in a specific direction, either horizontal or vertical.

Algorithm: Density Cal.

Input: All net segments in a direction, $Segs$
Output: Global and detailed routing densities, D_G and D_L in the specified direction of each grid cell

1. Initialize $D_G(g)$ and $D_L(g)$ to 0 and $I(g)$, the interval tree, to an empty tree for each grid cell g
2. While $Segs$ is not empty
   3. Remove the first element s from $Segs$
   4. For each grid cell g that s completely passes through
      5. $D_G(g)=D_G(g)+1$
   6. For the grid cells g that s partially passes through
      7. Add the interval within g where s exists to $I(g)$
8. For each grid cell g
   9. Assign $D_L(g)=max(scan\text{-}line\ of\ I(g))$

tree $I(g)$ is set for grid g (line 1 of Algorithm Density Cal.). For example, the global and detailed densities of the

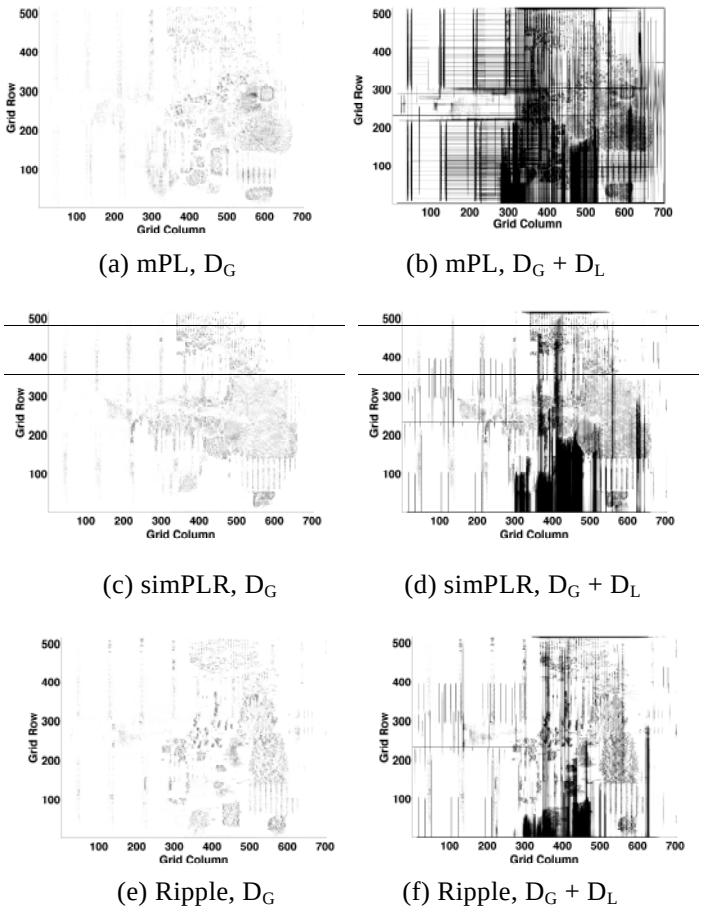
grid cell given in **Figure 6(a)** are shown in **Figures 6(b)** and **6(c)**, respectively.

First element s of the set $Segs(g)$, which is a degree-two net segment, is considered. If s completely passes through grid cell g, then the global density, D_G is added by one and s is removed from the set (lines 4 and 5 of Algorithm Density Cal.).

If s is a partial segment, then the interval within g where s exists is added to $I(g)$ (line 7 of Algorithm Density Cal.). The partial segments can be stored in an interval tree and the entire detailed component can be found by adding each segment in the grid cell to obtain a staircase-like function. The maximum number of routes that are hidden in a global cell is the local density and is found by using a scan-line algorithm. The sum of the two components is the *tree* routing density of each grid cell.

As an example, global and local densities for the gird cell in **Figure 6(a)** are shown in **Figures 6(b**) and **6(c)**. In these figures, the green lines show the global routes and the red dashed lines show the routes that are partially or entirely inside a grid cell. As it can be seen in these figures, the global density $D_G=4$ and the detailed density $D_L=4$ which means that the routing requirement for this grid cell is equal to 8, not 4 as it would be in global routing calculations. In **Table 4**, the global density and the percentage increase when both global and detailed densities are considered for the ISPD2011 circuits are given. From this table, it can be seen that the density increases significantly when detailed routing components are also considered.

In Figures 8(a) to **(f)**, the overflow for circuit Superblue1 from ISPD2011 benchmarks, when only the global routes are considered and then when both the global and detailed routing components are considered, are shown for three placers mPL11, Ripple and SimPLR. These figures indicate that the true overflow is masked during the global routing. However, using the proposed algorithm, Density Cal., the real amount of overflow can be easily calculated and provisions for solving the problem can devised both during placement and global routing.



(a) mPL, D_G (b) mPL, $D_G + D_L$

(c) simPLR, D_G (d) simPLR, $D_G + D_L$

(e) Ripple, D_G (f) Ripple, $D_G + D_L$

Figure 8: Overflow for circuit Superblue1 from ISPD2011 benchmarks before and after considering detailed routing density.

4.2.Via and Double Via Considerations

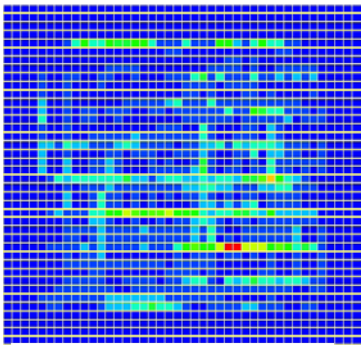
Another issue with removing nets is that even though the capacity of the edges can be adjusted to compensate for the local nets, the vias that they can use are not considered. This problem is intensified when some vias need to be

Table 4: Average Detailed and Global Density of ISPD2011 circuits

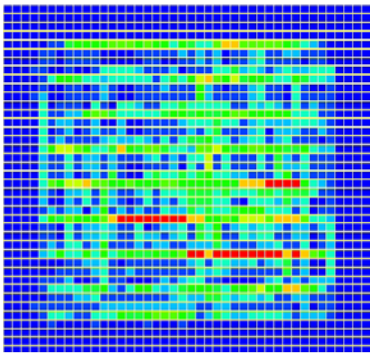
Circuit	% increase in routing density					
	mPL11		SimPLR		Ripple	
	D_G	% increase $D_G + D_L$	D_G	% increase $D_G + D_L$	D_G	% increase $D_G + D_L$
sb1	6586043	297.10	6705217	86.22	6387276	75.33
sb2	15503042	237.04	15419724	71.31	15885785	72.33
sb4	4230741	322.74	5178469	143.20	4454777	168.00
sb5	8074984	372.06	8507601	92.72	8029904	95.83
sb12	7589563	81.74	9214390	50.63	9822201	47.71
sb15	7045965	89.75	7583768	81.65	7837951	86.56
sb18	3765783	156.50	4784412	40.38	5441054	39.10

double vias. At current process nodes, double vias are required to ensure signal integrity and improve yield. Each double via uses two or more tracks, which will reduce the resources that wires can use. Not considering the vias used by local nets and the double vias can result in under-estimating the congestion of the circuit.

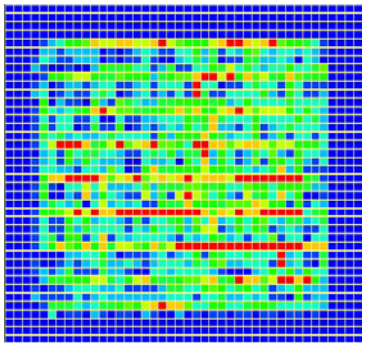
To show the gravity of this problem, an industry circuit in which double vias are used is tested. In **Figure 9**, the congestion of this industry circuit is shown. In **Figure 9(a),** the traditional congestion when local nets and accurate density are not considered, is shown. In **Figures 9(b)** and **9(c)**, when local nets are considered and double vias are added to the calculations are shown. It can be seen that not considering the congestion produced by the local nets and size of vias can result in under-estimating the congestion and in some cases can result in unroutable circuits. The data



(a) Traditional



(b) Using Algorithm Density



(c) Using Algorithm Density Cal. with double

Figure 9: Congestion using different techniques.

for the pin density and the number of tracks needed are given in **Table 5**. In this table, Column 1 is the region location of the circuit and Column 2 is number of tracks needed for routing while not accounting for double via sizes. In Column 3, the number of tracks when double via sizes are accounted for are given. It can be seen that the number of tracks needed are increased by 10% which is a large increase for this circuit with only 2000 standard cells. This problem is only intensified as the circuit sizes become bigger.

5. CONCLUSIONS

As circuits are becoming more complicated, simple goals such as reducing total wire length are replaced with more sophisticated measures such as routability. However, the main objective of the placement has remained reducing total HPWL.

In this paper, several shortcomings of the current wire length estimation technique are discussed, and low cost techniques to improve the accuracy and fidelity of the estimations have been proposed. Furthermore, considerations for detailed routing are discussed and an

Table 5: The number of tracks needed when double vias are considered.

region	single vias # tracks	double vias # tracks
C region: 1	6	7
C region: 3	10	11
C region: 5	14	15
C region: 7	10	11
C region: 9	15	16
C region: 11	11	13
C region: 13	9	10
C region: 15	9	10
C region: 17	10	11
R region: 18	1	2
C region: 19	10	10
C region: 21	13	15
C region: 23	10	12
R region: 24	1	2
C region: 25	10	12
C region: 27	10	11
C region: 29	11	13
C region: 31	10	11
C region: 33	10	11
C region: 35	4	5
I region: 40	2	4
I region: 41	1	2
I region: 42	2	4
Core tracks	182	204

algorithm is proposed to account for pin density.

REFERENCES

[1] N. Viswanathan, C. Alpert, C. Sze, Z. Li, G. Nam, and J. Roy. ISPD 2011 routability-driven placement contest and benchmark suite, www.ispd.cc/contests/11/ispd2011 contest.html, 2011.

[2] M. Kim, J. Hu, D. Lee, and I. Markov. A simPLR method for routability-driven placement. In Proc. of ICCAD, pages 67–73, 2011.

[3] X. He, T. Huang, L. Xiao, H. Tian, G. Cui, and E. Young. Ripple: an effective routability-driven placer by iterative cell movement. In Proc. of ICCAD, pages 74–70, 2011.

[4] M. Hsu, S. Chou, T. Lin, and Y. Chang. Routability driven analytical placement for mixed-size circuit designs. In Proc. of ICCAD, pages 80–84, 2011.

[5] A. Kahng, J. Lienig, I. Markov, and J. Hu. VLSI Physical Design: From Graph Partitioning to Timing Closure. Springer, 2011.

[6] C. Alpert, Z. Li, M. Moffitt, G.-J. Nam, J. Roy, and G. Tellez. What makes a design difficult to route. In Proc. of ICCAD, pages 7–12, 2010.

[7] C. Alpert and G. Tellez. The importance of routing congestion analysis. DAC.COM Knowledge Center Article, pages 1–14, 2010.

[8] P. Madden. Reporting of standard cell placement results. In Proc. of ISPD, pages 30–35, 2001.

[9] A. Kahng and G. Robins. A new class of iterative Steiner tree heuristics with good performance. IEEE Trans. on CAD, 11(7):893–902, 1992.

[10] C. Chu. Flute: fast lookup table based wirelength estimation technique. In Proc. of ICCAD, pages 696–701, 2004.

[11] A. Davoodi, V. Khandelwal, and A. Srivastava. Empirical models for net-length probability distribution and applications. IEEE Trans. on VLSI, 12:1066–1075, October 2004.

[12] J. Roy and I. Markov. Seeing the forest and the trees: Steiner wirelength optimization in placement. IEEE Transactions on CAD, 26(4):632 –644, Apr. 2007.

[13] T. Jindal, C. Alpert, J. Hu, Z. Li, G. Nam, and C. Winn. Detecting tangled logic structures in vlsi netlists. In Proc. of DAC, pages 603–608, 2010.

[14] C.-H. Hsu, H.-Y. Chen, and Y.-W. Chang. Multi-layer global routing considering via and wire capacities. In ICCAD, pages 350–355, 2008.

[15] T.-H. Wu, A. Davoodi, and J. T. Linderoth. GRIP: scalable 3D global routing using integer programming. In Proc. of DAC, pages 320–325, 2009.

[16] S. Bodapati and F. Najm. Prelayout estimation of individual wire lengths. IEEE Trans. on VLSI, 9(6):943–958, 2001

[17] A. Kahng and S. Reda. Intrinsic shortest path length: a new, accurate a priori wirelength estimator. In Proc. of ICCAD, volume 2005, pages 173–180, 2005.

[18] B. Fathi, L. Behjat, and L. Rakai. A pre-placement net length estimation technique for mixed-size circuits. In Proc. of SLIP, pages 45–52, 2009.

[19] A. Farshidi, L. Behjat, L. Rakai, and B. Fathi. A pre-placement individual net length estimation model and an application for modern circuits. Integration, 44(2):111–122, 2011.

[20] H. Javaid, A. Ignjatovic, and S. Parameswaran. Fidelity metrics for estimation models. In Proc. of the ICCAD, pages 1–8, 2010.

[21] J. Rodgers and W. Nicewander. Thirteen ways to look at the correlation coefficient. The American Statistician, 42(1):59–66, 1988.

[22] S. Adya, S. Chaturvedi, J. Roy, D. Papa, and I. Markov. Unification of partitioning, floorplanning and placement. In Proc. of ICCAD, pages 550–557, 2004.

[23] Yaoguang Wei, Cliff Sze, Natarajan Viswanathan, Zhuo Li, Charles J. Alpert, Lakshmi Reddy, Andrew D. Huber, Gustavo E. Tellez, Douglas Keller, and Sachin S. Sapatnekar. 2012. GLARE: global and local wiring aware routability evaluation. In Proc. of DAC, pages 768-773, 2012.